

REMARKS

Applicant and applicant's attorney express appreciation to the Examiner for the courtesies extended during the recent telephone interview held on January 17, 2003. The claim amendments made by this paper are consistent with the proposals discussed during the interview.

In the Office Action mailed September 30, 2002, all claims (1-26) were rejected under 35 U.S.C. § 103(a) as being obvious in view of U.S. Patent No. 5,524,238, to Miller and U.S. Patent No. 5,893,128, to Nauckhoff. The oath and declaration was also declared defective. A new oath and declaration that is in compliance with 37 C.F.R. 1.67(a) has been provided with this Amendment. By this paper, claims 1, 5-6, 9-13, 17-19, 21 and 25 have been amended and claim 26 has been cancelled. Accordingly, claims 1-25 now remain pending.

As discussed during the Interview, and as recited in the claims, the present invention is generally directed to a method for allowing client applications to control how a particular high-level document command is implemented in a database management system. The claimed method comprises various acts, including: an act of receiving a high-level document command, acts of identifying and notifying one or more client applications of the receipt of the high-level document command prior to implementing the high-level document command, and acts of receiving and following instructions from the one or more client applications on how to affect the implementation of the high-level document command.

As further discussed during the interview, Miller does not teach of a method for allowing client applications to control how commands are implemented and that includes the acts of receiving a command, the acts of identifying and notifying the applications of the receipt of the command prior to implementing the command, and that further includes the acts of receiving and following instructions from the applications on how to affect the implementation of the command.

Rather, Miller generally discloses a method for utilizing an interface to improve user productivity. The interface disclosed in Miller stores a plurality of record files associated with specific users by user ID that can be referenced when the user logs in. Col. 2, ll. 8-13, Col. 10, 25-27. The usefulness of the invention is explained with relation to the example provided in Col. 9-10. As described, the utility of Miller is the enabling of the intelligent interface to automatically populate data entry fields with appropriate data from stored tables and files of the interface that associate certain data fields once the user has logged on. (see Col. 9, line 63 thru Col. 11, line 33.)

Miller does mention that certain input and output commands may be intercepted and that functions may be performed on the data before passing the data on to an application. As stated in Miller, the system "performs a defined set of functions on the data before passing the data on or passing other data on to the user or application." Col. 2, ll. 59-62. However, this is distinguished from the present invention in which the third party applications are notified of commands prior to their implementation to allow the applications to affect the implementation of the commands. See the amended claims and the Specification page 14, ll. 3-6.

In the Office Action, it is stated that the admitted prior art discloses an act of notifying applications that a document command has been received (referring to Fig. 1, and page 4, lines 1-10). However, this is not the case. More particularly, the admitted prior art that is disclosed in the specification correctly states that in existing systems, "Once the high-level document command is **implemented**, the database management system may notify other client applications using notification module 150...Then, the other client applications may perform a variety of processes in response to that notification..." (emphasis added, p. 4, ll. 3-6.) This is distinguished from the acts disclosed in the claimed method for at least the foregoing reasons. In particular,

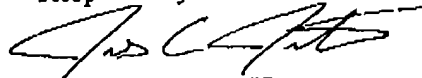
the amended claims recite that the acts of identifying and notifying the applications of the receipt of the command are performed **prior** to implementing the command. Accordingly these recited acts in combination with the other recited acts distinguish the Miller reference and the admitted prior art from the amended claims.

For at least the foregoing reasons, claims 1-25 should now be considered in condition for allowance.

In the event that the Examiner finds remaining impediment to a prompt allowance of this application that may be clarified through a telephone interview, the Examiner is requested to contact the undersigned attorney.

Dated this 21st day of January, 2003.

Respectfully submitted,



JENS C. JENKINS

Registration No. 44,803

RICK D. NYDEGGER

Registration No. 28,651

Attorneys for Applicant



022913

PATENT TRADEMARK OFFICE

VERSION WITH MARKINGS TO SHOW CHANGES MADE
(Serial No. 09/680,118)

In the claims:

1. (Amended) In a database management system that includes a database engine that [accesses and updates objects in a database, the database engine receiving] receives and implements high-level document commands, each high-level document command comprising one or more operations to be performed [for performing an operation] on a document [that is associated with a plurality of tables in the database], a method for allowing client applications to control how a particular high-level document command is implemented [in the database], the method comprising the following:

an act of receiving a high-level document command meeting certain criteria;

prior to implementing the high-level document command, an act of identifying one or more client applications that are to be notified of the receipt [implementation] of the high-level document command;

prior to implementing the high-level document command, an act of notifying the one or more identified client applications that the [a] high-level document command meeting the certain criteria has been received;

an act of receiving instructions from the one or more client applications on how to affect the implementation of the high-level document command [in the database]; and

an act of following the received instructions when implementing the high-level document command for performing the one or more operations on the document, or not implementing the high-level document command at all if the received instructions so indicate.

5. (Amended) The method in accordance with Claim 1, wherein the received instructions are for changing how the high-level document command is to be implemented in [the] a database that is accessed by the database management system.

6. (Amended) The method in accordance with Claim 1, wherein the received instructions are for preventing the high-level document command from being implemented at all [in the database].

VERSION WITH MARKINGS TO SHOW CHANGES MADE
(Serial No. 09/680,118)

9. (Amended) The method in accordance with Claim 1, wherein the high-level document command is for moving [a] the document.
10. (Amended) The method in accordance with Claim 1, wherein the high-level document command is for deleting [a] the document.
11. (Amended) The method in accordance with Claim 1, wherein the high-level document command is for copying [a] the document.
12. (Amended) The method in accordance with Claim 1, wherein the high-level document command is for updating [a] the document.
13. (Amended) The method in accordance with Claim 1, wherein the high-level document command is for adding [a] the document.

VERSION WITH MARKINGS TO SHOW CHANGES MADE
(Serial No. 09/680,118)

17. (Amended) In a database management system that includes a database engine that accesses and updates objects in a database, the database engine receiving high-level document commands, each high-level document command for performing an operation on a document that is associated with a plurality of tables in the database, a method for allowing client applications to control how a particular high-level document command is implemented in the database, the method comprising the following:

an act of receiving a high-level document command meeting certain criteria; and

a step for allowing one or more client applications to affect how the received high-level document command is to be implemented, if at all, in the database,

the step for allowing one or more client applications to affect how the received high-level document command is to be implemented, including an act of identifying one or more client applications that are to be notified of the receipt of the high-level document command, and an act of notifying the one or more identified client applications that a high-level document command meeting certain criteria has been received prior to implementing the high-level document command.

VERSION WITH MARKINGS TO SHOW CHANGES MADE
(Serial No. 09/680,118)

18. (Amended) The method in accordance with Claim 17, wherein the step for allowing one or more client applications to affect how the received high-level document command is to be implemented further includes: [comprises the following:

an act of identifying one or more client applications that are to be notified of the implementation of the high-level document command;

an act of notifying the one or more identified client applications that a high-level document command meeting certain criteria has been received;]

an act of receiving instructions from the one or more client applications on how to affect the implementation of the high-level document command in the database; and

an act of following the received instructions when implementing the high-level document command, or not implementing the high-level document command at all if the received instructions so indicate.

VERSION WITH MARKINGS TO SHOW CHANGES MADE
(Serial No. 09/680,118)

19. (Amended) A computer program product for use in a database management system that includes a database engine that [accesses and updates objects in a database, the database engine receiving] receives and implements high-level document commands, each high-level document command comprising one or more operations to be performed [or performing an operation] on a document [that is associated with a plurality of tables in the database], a computer program product for implementing a method for allowing client applications to control how a particular high-level document command is implemented [in the database], the computer-program product comprising a computer-readable medium that contains computer-executable instructions for performing the following:

an act of detecting the receipt of a high-level document command meeting certain criteria;

prior to implementing the high-level document command, an act of identifying one or more client applications that are to be notified of the receipt [implementation] of the high-level document command;

prior to implementing the high-level document command, an act of causing the one or more identified client applications to be notified that the [a] high-level document command meeting certain criteria has been received;

an act of detecting the receipt of instructions from the one or more client applications on how to affect the implementation of the high-level document command [in the database]; and

an act of following the received instructions when implementing the high-level document command for performing the one or more operations on the document, or not implementing the high-level document command at all if the received instructions so indicate.

VERSION WITH MARKINGS TO SHOW CHANGES MADE
(Serial No. 09/680,118)

21. (Amended) The computer program product in accordance with Claim 20, wherein the computer-executable instructions for performing additional high-level document commands comprise computer-exccutable instructions for atomically implementing the additional high-level document commands and the received high-level document command in [the] a database accessed by the database management system.

25. (Amended) A database management system for implementing high-level document commands for performing an operation on a document, each document being associated with a plurality of tables in an underlying database, the database management system comprising:

a database application that is configured to send high-level document commands;

a notification component that is configured to send a notification to any identified client application when a given high-level document command is received by the database management system, and prior to implementation of the high-level document command;

an instruction receiver module that is configured to receive instructions from the notified third party application on how to implement the high-level document command; and

a database engine configured to follow the received instructions when implementing the high-level document command.

VERSION WITH MARKINGS TO SHOW CHANGES MADE
(Serial No. 09/680,118)

In the Specification:

The paragraph beginning on line 9 of page 16 has been amended as follows:

When performing additional high-level document commands over and above the received high-level document commands, the document commands may be implemented as a "group operation" in which all of the document commands are implemented in a single transaction, all of the high-level document commands being implemented, or none at all. Group operations are described in co-pending United States applications serial number 09/680,104, entitled "METHODS AND SYSTEMS FOR PERFORMING LONG-RUNNING ATOMIC GROUP OF HIGH-LEVEL OPERATIONS OVER A DATABASE SYSTEM WITHOUT LONG-RUNNING TRANSACTION SUPPORT," filed October 4, 2000, and which is incorporated herein by reference in its entirety. [[NOT YET ASSIGNED - ATTORNEY DOCKET NUMBER 13768.171], filed on the same date herewith, and entitled "METHODS AND SYSTEMS FOR PERFORMING HIGH-LEVEL GROUP OPERATIONS IN A DATABASE MANAGEMENT SYSTEM", which is incorporated herein by reference in its entirety.]